

**Problema 2: ramen**

**100 de puncte**

Ai deschis recent un restaurant cu specific japonez, iar lucrurile nu merg grozav. Uneori clienții ajung să aștepte foarte mult mâncarea comandată, iar acum crezi că ai înțeles de ce se întâmplă acest lucru.

Restaurantul nu are mese, ci un singur bar foarte lung dotat cu o bandă rulantă care transportă porțiile de mâncare de la bucătărie la client. Barul are **500.000.000** de scaune numerotate în ordine crescătoare, scaunul **1** fiind cel mai apropiat de bucătărie. Uneori clienții fac noi comenzi. O comandă făcută la secunda **T** de către clientul aflat pe scaunul cu numărul **P** va ajunge instant la bucătărie. Prepararea mâncării va dura **D** secunde, iar apoi mâncarea va fi pusă pe bandă și va dura exact **P** secunde ca aceasta să ajungă la client. În acest timp, mâncarea va trece prin fața scaunelor **1, 2, ... P - 1**. Dacă dintr-un anumit motiv clientul nu își ridică mâncarea de pe bandă, aceasta va continua să se deplaseze. În caz contrar, clientul în cauză se așteaptă ca mâncarea să ajungă la scaunul său la secunda **T + D + P**.

Deocamdată restaurantul servește un singur fel de mâncare: *ramen*. Astfel, comenzile făcute de clienți ajung să fie ușor interschimbabile, iar aceștia se arată foarte deschiși la a profita de pe urma acestui fapt.

Se cunosc următoarele:

- Un client poate avea zero sau mai multe comenzi în așteptare.
- Un client care are zero comenzi în așteptare este complet inactiv.
- Numărul de comenzi în așteptare ale unui client care face o comandă la secunda **T** va crește cu o unitate exact la secunda **T**.
- Un client care are în așteptare cel puțin o comandă va ridica de pe bandă prima porție de ramen care trece prin fața sa, indiferent dacă aceasta îi era destinată sau nu. Dacă va face acest lucru la momentul **T**, numărul său de comenzi în așteptare va scădea cu o unitate exact la momentul **T**.

Spre exemplu, analizăm situația următoare cu două comenzi:

Durata de preparare a ramenului este **D = 2** secunde. Această valoare este o constantă și se aplică identic fiecărei comenzi.

La secunda **T1 = 10**, persoana de pe scaunul cu numărul **P1 = 8** (să o numim **A**) comandă o porție de ramen. La secunda **T1 + D = 12**, porția sa este pusă pe bandă.

La secunda **T2 = 16**, persoana de pe scaunul cu numărul **P2 = 6** (să o numim **B**) comandă o porție de ramen. La secunda **T2 + D = 18**, porția sa este pusă pe bandă.

La secunda **19** porția destinată clientului **A** trece prin fața scaunului **6**, iar clientul **B**, fiind el însuși în așteptarea unei comenzi o va lua și o va mânca. El va mânca deci la secunda **19** și va ignora apoi propria sa comandă, care va trece pe lângă el.

La secunda **28** porția destinată clientului **B** va ajunge la clientul **A**, iar acesta o va lua și o va mânca. El va mânca deci la secunda **28**.

Se observă că în general, în ciuda întârzierilor create, fiecare client va consuma exact câte porții a comandat.

**Cerințe**

Pentru a evalua impactul acestui obicei asupra timpilor de așteptare, ai obținut date despre toate comenzile date în ziua curentă. Îți propui să afli, pentru fiecare comandă următoarea valoare: dacă respectiva comandă este a **NR**-a făcută de clientul respectiv, care este secunda la care clientul în cauză va mânca pentru a **NR**-a oară?

**Date de intrare**

Fișierul de intrare **ramen.in** va conține pe prima linie numărul de comenzi **N**, respectiv timpul de preparare a unei porții de ramen **D**. Următoarele **N** linii vor conține câte o pereche de numere: **T[i]**, secunda la care este făcută a **i**-a comandă, respectiv **P[i]**, numărul scaunului de la care s-a făcut a **i**-a comandă. Se garantează că timpii de comandă sunt distincți și sunt crescători în ordinea citirii lor.

### Date de ieșire

Fișierul de ieșire `ramen.out` va conține  $N$  linii, fiecare conținând o singură valoare naturală: a  $i$ -a valoare va reprezenta răspunsul cerut pentru a  $i$ -a comandă în ordinea citirii.

### Restricții și precizări

1.  $1 \leq N \leq 100.000$
2.  $0 \leq D, T[i] \leq 500.000.000$ , pentru orice  $1 \leq i \leq N$
3.  $1 \leq P[i] \leq 500.000.000$ , pentru orice  $1 \leq i \leq N$
4. Se garantează că  $T[i] < T[i + 1]$ , pentru orice  $1 \leq i < N$ .
5. Pentru teste în valoare totală de 22 de puncte, se garantează în plus față de restricțiile generale că  $N \leq 2000$  și  $D, T[i], P[i] \leq 5000$
6. Pentru alte teste în valoare totală de 25 de puncte, se garantează în plus față de restricțiile generale că  $N \leq 2000$ .

### Exemple

ramen.in	ramen.out	Explicații și comentarii
2 2 10 8 16 6	28 19	Exemplul descris în enunț.
3 2 5 4 6 4 7 3	12 13 10	Observați că în acest exemplu clientul de pe scaunul cu numărul 4 face două comenzi. Răspunsul corespunzător primei comenzi este secunda în care clientul mănâncă pentru prima oară, iar răspunsul corespunzător comenzii cu numărul doi este secunda în care clientul mănâncă pentru a doua oară.
3 0 0 6 3 3 4 5	10 3 8	Observați că în acest exemplu clientul de pe scaunul cu numărul 3 face o comandă la secunda 3. Exact în același moment, îi apare o porție de ramen în față, iar el o consumă imediat.

Timp maxim de execuție/test: 1,8 sec

Memorie totală: 128 MB din care 32 MB pentru stiva

Dimensiunea maximă a sursei: 20 KB